Data Hiding in Image and Video: Part I—Fundamental Issues and Solutions

Min Wu, *Member, IEEE,* and Bede Liu, *Fellow, IEEE*

***Abstract*—In this Part I of a two-part paper, we address a number of fundamental issues of data hiding in image and video and propose general solutions to them. We begin with a review of two major types of embedding, based on which we propose a new multilevel embedding framework to allow the amount of extractable data to be adaptive according to the actual noise condition. We then study the issues of hiding multiple bits through a comparison of various modulation and multiplexing techniques. Finally, the nonstationary nature of visual signals leads to highly uneven distribution of embedding capacity and causes difficulty in data hiding. We propose an adaptive solution switching between using constant embedding rate with shuffling and using variable embedding rate with embedded control bits. We verify the effectiveness of our proposed solutions through analysis and simulation. And Part II [1] will apply these solutions to specific design problems for embedding data in grayscale and color images and video.**

***Index Terms*—Data hiding, digital watermarking, embedding capacity, modulation and multiplexing, shuffle.**

I. Introduction

In THE recent decade, new devices and powerful software have made it possible for consumers worldwide to access, create, and manipulate multimedia data. Internet and wireless networks offer ubiquitous channels to deliver and to exchange such multimedia information. However, the potential offered by the information technology era cannot be fully realized without the guarantee on the security and protection of multimedia data. Digital watermarking and data hiding[1] are schemes to embed secondary data in digital media [2]–[18] for a variety of applications, including ownership protection, authentication, access control, and annotation. Data hiding is also found to be useful to send side information in multimedia communication for achieving additional functionalities or enhancing performance [3].

In this paper, we address both fundamental and design issues regarding data hiding in image and video. The paper is organized into two parts. Part I addresses several fundamental issues and proposes general solutions. Based on Part I, Part II presents new data hiding algorithms and system designs for image and video [1].

Data hiding can be considered as a communication problem where the embedded data is the signal to be transmitted. A fundamental problem is the embedding capacity. That is, how many bits can be embedded in a host signal. The answer depends on the required robustness. Earlier works regarding the embedding capacity focused on spread spectrum additive watermarking, by which a noise-like watermark is added to a host image and is later detected via a correlator [19], [20]. This embedding can be modeled as communication over a channel with additive white gaussian noise (AWGN) [21], [22]. Other researchers studied the bounds of embedding capacity under blind detection [23]–[25]. Zero-error capacity has been studied for a watermark-based authentication system under magnitude-bounded noise [17], using the principles originally proposed by Shannon [26], [27]. In [28], Costa showed that the channel capacity under two additive Gaussian noises with one known to the sender equals to the capacity in the absence of the known noise. This result has been incorporated in information theoretical formulations of data hiding [29], [30].

The gap between the theoretical embedding capacity in data hiding and what is achievable in practice can be bridged by investigation of such issues as basic embedding mechanisms for embedding one bit and modulation/ multiplexing techniques for embedding multiple bits. The following problems require particular attention:

- *Distortion*: The distortion introduced by watermarking must be imperceptibly small for commercial or artistic reasons. However, an adversary intending to obliterate the watermark may be willing to tolerate certain degree of visible artifacts. Therefore, the distortions by embedding and by attack are often asymmetric, leading to a wide range of possible watermark-to-noise ratio.
- *Actual noise conditions*: An embedding system is generally designed to survive certain noise conditions. The watermarked data may encounter a variety of legitimate processing and malicious attacks, so the actual noise can vary significantly. Targeting conservatively at surviving severe noise would lead to the waste of actual payload, while targeting aggressively at light noise could result in the corruption of embedded bits. In addition, some bits, such as the ownership information and control information, are required to be more robust.
- *Uneven distribution of embedding capability*: The amount of data that can be embedded often vary widely from region to region in image and video. This uneven embedding capacity causes serious difficulty to high-rate embedding.

Manuscript received February 4, 2002; revised November 22, 2002. This work was supported in part by a R&D Excellence Grant from the State of New Jersey and by the National Science Foundation CAREER Award CCR-0133704. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Bruno Carpentieri.

M. Wu is with the Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742, USA, minwu@eng.umd.edu.

B. Liu is with Department of Electrical Engineering, Princeton University, Princeton, NJ 08544, USA, liu@ee.princeton.edu.

Digital Object Identifier 10.1109/TIP.2003.810588

[1]The two terms *data hiding* and *digital watermarking* will be used interchangeably in this paper.

1057-7149/03$17.00 © 2003 IEEE

We present in Section II of this Part I a general framework and a layered view applicable to general data hiding problems. We also summarize the robustness versus capacity tradeoff for two major types of embedding and identify their relative advantages. This serves as a basis for a new paradigm of *multilevel data hiding* presented in Section III. Then in Section IV, we compare the applicability, advantages, and limitations of four general modulation/multiplexing techniques used for embedding multiple bits. And in Section V, we propose a comprehensive solution to the uneven embedding capacity problem. Finally, we summarize Part I in Section VI.

II. Preliminaries

In this section, we review a few concepts and principles of data hiding that will be used throughout the discussion in this paper.

A. A Data Hiding Framework

A typical data hiding framework is illustrated in Fig. 1. Starting with an original digital media (I_0), which is also known as the *host media* or *cover media*, the embedding module inserts in it a set of secondary data ($\underline{b}$), which is referred to as *embedded data* or *watermark*, to obtain the *marked media* (I_1). The insertion or embedding is done such that I_1 is perceptually identical to I_0. The difference between I_1 and I_0 is the distortion introduced by the embedding process. In most cases, the embedded data is a collection of bits, which may come from an encoded character string, from a pattern, or from some executable agents, depending on the application.

The embedded data $\underline{b}$ will be extracted from the marked media I_1 by a detector, often after I_1 has gone through various processing and attacks. The input to the detector is referred to as *test media* (I_2), and the *extracted data* from I_2 is denoted by $\hat{\underline{b}}$. The difference between I_2 and I_1 is called *noise*. In such applications as ownership protection, fingerprinting, and access control, accurate decoding of hidden data from distorted test media is preferred. In other applications such as authentication and annotation, robustness is not critical.

The key elements in many data hiding systems include

- a perceptual model that ensures imperceptibility;
- a mechanism for embedding one bit;
- techniques for embedding multiple bits via appropriate modulation/multiplexing;
- what data to embed;
- how to handle the parts of host media in which it is difficult to embed data;
- how to enhance robustness and security.

We can view these elements through a layered structure shown in Fig. 2, analogous to that in communications. The lower layers deal with how one or multiple bits are embedded imperceptibly in the host media. Upper layers for achieving additional functionalities can be built on top of these lower layers.

B. Two Basic Embedding Mechanisms

The embedding of one bit in host media is basic to every data hiding system. Almost all embedding approaches belong

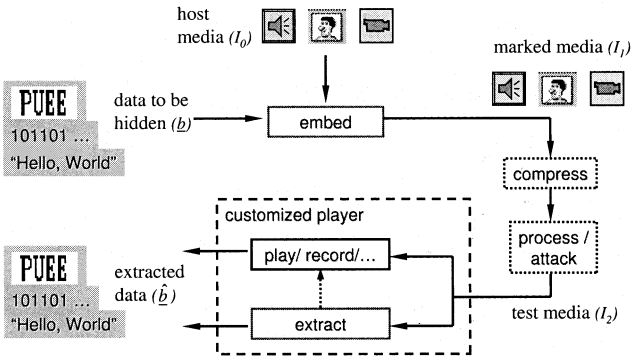

Fig. 1. General framework of data hiding systems.

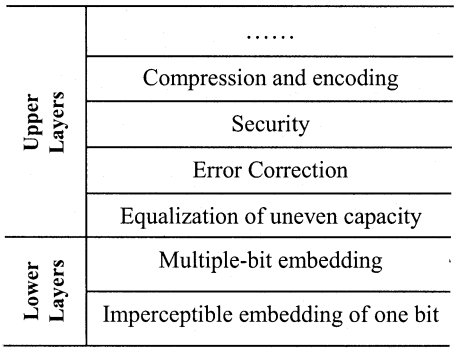

Fig. 2. Layered structure of data hiding.

to one of two general types, which was proposed independently in [29]–[31].

In *Type-I* embedding, the secondary data, possibly encoded, modulated, and/or scaled, is added to the host signal, as illustrated in Fig. 3(a). The addition can be performed in a specific domain or on specific features. To embed one bit b, the difference between marked signal I_1 and the original host signal I_0 is a function of b, i.e., $I_1 - I_0 = f(b)$. I_0 can be a major noise source in detection. Although it is possible to detect b directly from I_1 [32], the knowledge of I_0 will enhance detection performance by eliminating the interference. Additive spread spectrum watermarking [20], [33] is an example of Type-I.

In *Type-II* embedding, the signal space is partitioned into subsets, each of which is mapped by a function $g(\cdot)$ to the set of values taken by the secondary data, as illustrated in Fig. 3(b). The marked value I_1 is then chosen from the subset that maps to b, so that the relationship of $b = g(I_1)$ is deterministically enforced. To minimize perceptual distortion, I_1 should be as close to I_0 as possible. A simple example is *odd-even embedding*, whereby a closest even number is used as I_1 to embed a "0" and a closest odd number is used to embed a "1." The embedded bit is extracted simply by checking the odd-even parity,[2] which does not require the knowledge of original I_0. There can be other constraints imposed on I_1 for robustness considerations. For example, the enforcement can be done in a quantized domain with

[2]Odd-even embedding is *not* equivalent to replacing the least-significant-bit (LSB) with the data to be embedded, because LSB embedding does not always produce the closest I_1 to satisfy the relationship $b = g(I_1)$. If the probabilistic distribution of I_0 in each quantization interval of size Q can be approximated by a uniform distribution, the MSE of odd-even embedding is $Q^2/3$ while the embedding by replacing LSB is $7Q^2/12$.

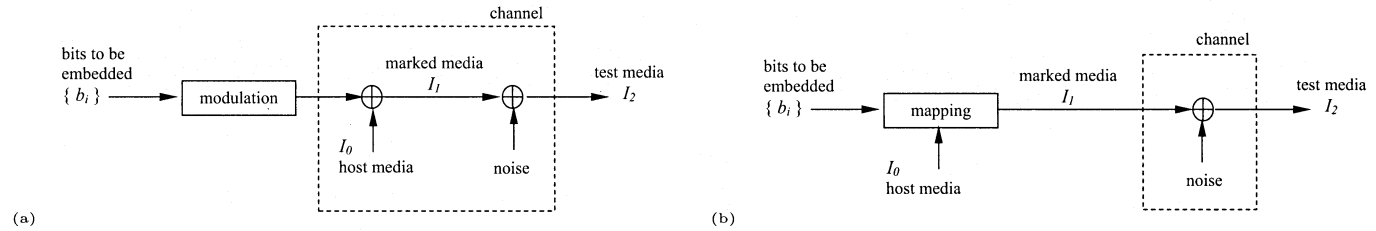

Fig. 3. Channel models for Type-I (a) and Type-II (b) embedding.

a step size Q [18], [31]. The odd-even enforcement can also be extended to higher dimensions involving features computed from a set of host components.

The odd-even embedding can be viewed as a special case of the table-lookup embedding, which can provide an additional level of security by using a random lookup table as the mapping $g(\cdot)$ [34], [35]. There are many other ways to partition the signal space. For example, for a pair of host samples or coefficients v_1 and v_2, we may generate the marked version v'_1 and v'_2 so that a "1" is embedded by forcing $v'_1 > v'_2$ and a "0" is embedded by forcing $v'_1 \leq v'_2$ [36]. We can also use enforcing signs to embed one bit [37], [38]. Many schemes that use noncoherent detection[3] belong to this Type-II category. It is the deterministically enforced relationship on I_1 that removes the need of using the host signal I_0. For convenience, we shall refer the collection of image pixels or coefficients on which the relation is enforced as an *embedding unit*. If the enforcement is performed on a quantity derived from the embedding unit, such as the sum of a few coefficients and the signs of a coefficient, we shall refer to the quantity as a *feature*.

C. Capacity Comparison for Type-I and Type-II

We compare the capacities versus watermark-to-noise ratio of Type-I and Type-II embedding under blind detection. Specifically, we fix the mean squared error introduced by embedding to be E^2 and model the channels as the follows. For Type-I, we consider a Continuous-Input-Continuous-Output (CICO) channel model. We assume that the AWGN noise consists of Gaussian processing noise with variance σ^2 and host interference with standard deviation 10 times the amplitude of the watermark signal (i.e., $\sigma_I = 10E$)[4]. For Type-II, we consider a Discrete-Input-Discrete-Output (DIDO) Binary-Symmetric-Channel model[5] for odd-even embedding with quantization step $Q = \sqrt{3}E$. The capacity for Type-I and Type-II under these assumptions are [18]

$$C_I = \frac{1}{2}\log_2\left(1+\frac{E^2}{(10E)^2+\sigma^2}\right) \quad (1)$$

$$C_{II} = 1 - h_{\min\{1/2,\, 2\sum_{k=0}^{+\infty} Q((4k+1)Q/2\sigma) - Q((4k+3)Q/2\sigma)\}} \quad (2)$$

[3]Non-coherent detection, also known as "blind detection," refers to the detection of the embedded data without the use of the original unwatermarked copy.

[4]In general, the magnitude ratio between the host signal and the watermark depends on the content of the host signal and human perceptual models [33]. A ratio around 10 is typical [20] and is adopted here. Small changes in the ratio will not lead to significant changes in the capacity curve.

[5]Other channel modeling for Type-II can be found in [18].

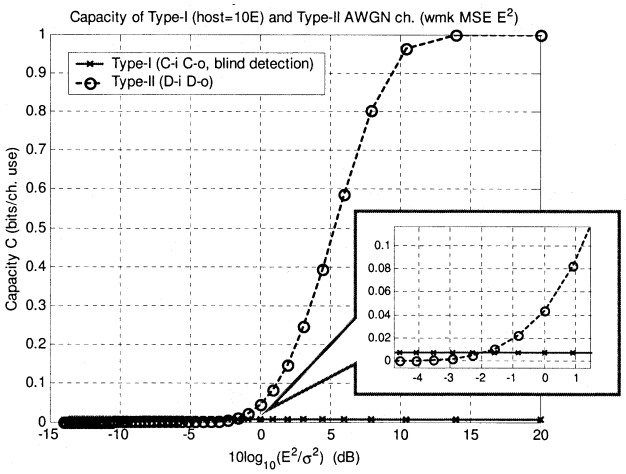

Fig. 4. Capacity of Type-I (CICO channel) and Type-II (DIDO channel) embedding under AWGN noise.

where h_p is the binary entropy

$$h_p = -p\cdot\log(p) - (1-p)\cdot\log(1-p).$$

The capacities versus the *watermark-to-noise ratio (WNR)*, E^2/σ^2, are plotted in Fig. 4. It is seen that the capacity of Type-II is much higher than that of Type-I until the WNR falls below 0 dB. Thus Type-II is useful under low noise condition while Type-I is suitable for strong noise, especially when the noise is stronger than the watermark.

Motivated by Costa's information theoretical result [28], distortion compensation has been proposed to be incorporated into quantization-based enforcement embedding [30], [39], [40], where the enforcement is combined linearly with the host signal to form a watermarked signal. The optimal scaling factor is a function of WNR and will increase the number of bits that can be embedded. This distortion compensated embedding can be viewed as a combination of Type-I and Type-II embedding. Among the practical embedding schemes, Type-II can reach comparable embedding rate to its distortion compensated counterpart at high WNR, while Type-I can reach similar rate to distortion compensated embedding at very low WNR.

III. Multilevel Embedding

An embedding scheme usually targets at a specific robustness level, leading to a specific total payload[6]. Fig. 5(a) shows

[6]The *total payload* is the total amount of data embedded in the host signal. It consists of the main *user payload* (such as ownership information and copy control policy) and any additional *control data* embedded to facilitate the data extraction.

the capacity versus WNR for Type-I in dashed line and that for Type-II in dotted line. The maximum number of bits that can be embedded reliably using Type-I or Type-II should follow the solid curve $C(x)$, which is the envelope of the two curves. For a watermark targeted to survive at a specific level of WNR, say x_1, the maximum number of payload bits that can be extracted reliably is $C(x_1)$ in Fig. 5(b), even if the actual WNR is higher than x_1. Thus the number of reliably extractable payload bits under different actual noise conditions follows the solid line $C_{I,1}(x)$ in Fig. 5(b), which is a step function with a jump at the design target WNR of x_1. If the design target WNR is different, say x_2, the number of reliably extractable payload bits would follow a different step function curve, $C_{I,2}(x)$. Therefore, using a single design target WNR will result in no extractable data when the actual noise is stronger than the design parameter, while the number of extractable bits does not increase even if the actual noise is weaker than that targeted in the design.

It is possible to use two targeted values of WNR in the design, so that a fraction α_1 of the embedded data survives a WNR of x_1, and all embedded data survives a higher WNR of x_2. The maximum number of extractable payload bit versus the actual noise conditions of this combined embedding would then follow a 2-step curve $C_{II}(x)$ in Fig. 5(c). This approach would allow more bits to be extractable than $C_{I,1}(x)$ when $x \geq x_2$, and than $C_{I,2}(x)$ when $x_1 < x < x_2$.

The above 2-level embedding can be extended to M-level embedding, by selecting M targeted WNR $[x_1, x_2, \ldots, x_{_M}]$ and the associated fraction $[\alpha_1, \alpha_2, \ldots, \alpha_M]$ where $x_1 < x_2 < \cdots < x_{_M}$ and $\sum_{i=1}^{M} \alpha_i = 1$, the maximum number of extractable bits $C_M(x)$ is

$$C_M(x)=\begin{cases} \sum_{i=1}^{M} \alpha_i C_{I,i}(x_i), & \text{if } x > x_{_M} \\ \sum_{i=1}^{k} \alpha_i C_{I,i}(x_i), & \text{if } x_k < x < x_{k+1},\ k=1,\ldots,M-1 \\ 0, & \text{if } x < x_1. \end{cases} \tag{3}$$

Let $\alpha_i = 1/M$, $x_{_L} = x_1$, $x_{_U} = x_{_M}$, and $x_{i+1} - x_i = (x_{_U} - x_{_L})/(M-1)$. For fixed $x_{_L}$ and $x_{_U}$, as M goes to infinity, we have

$$C_\infty(x) = \begin{cases} \frac{1}{x_{_U}-x_{_L}} \int_{x_{_L}}^{x_{_U}} C(t)dt, & \text{if } x > x_{_U} \\ \frac{1}{x_{_U}-x_{_L}} \int_{x_{_L}}^{x} C(t)dt, & \text{if } x_{_L} \leq x \leq x_{_U} \\ 0, & \text{if } x < x_{_L}. \end{cases} \tag{4}$$

This is illustrated in Fig. 5(d). We see that combining many embedding levels can achieve graceful degradation, so that the extractable information decays smoothly as the actual noise gets strong[7].

The graceful change of the amount of extractable information is desirable in many applications. The information to be embedded often requires unequal error protection (UEP). Some bits, such as the ownership information and control bits to facilitate the decoding of the actual payload bits, are required to be embedded more robustly than others. For access control or copy control applications in which a nontrivial number of bits are embedded in audio or video to indicate usage rules, these rules cannot be enforced until they are decoded. It is often desirable to enforce usage rules sooner on audio/video that are lightly compressed and have high commercial value. This can be realized by embedding the usage rules in multiple robustness levels. When the compression is light, the rules can be decoded by processing just a small segment of audio/video; and when the audio/video is heavily compressed, the rules can still be robustly decoded by processing a longer segment. In Part II, we will present data hiding algorithms and system designs for images and video using this multilevel embedding idea.

[7]In practice, both the fractions, $\{\alpha_i\}$, and the targeted WNR's, $\{x_i\}$, can be nonuniform to allow different emphasis toward different noise conditions.

IV. Techniques for Embedding Multiple Bits

Techniques that extend single-bit embedding to multiple-bit are evolved from modulation and multiplexing in classic communications [41]. They form an important element in data hiding systems for conveying both user payload and the side information that facilitates the extraction of user payload. In this section, we briefly review and compare four commonly used approaches: amplitude modulo modulation, orthogonal and biorthogonal modulation, time/spatial division modulation and multiplexing (TDM), and code division modulation and multiplexing (CDM). The comparison result is summarized in Table I.

A. Modulation and Multiplexing Techniques

Amplitude Modulo Modulation: Type-I embedding using antipodal or on-off modulation is a simple amplitude modulation. Under blind detection and perceptual constraints, it is uncommon in practice to use amplitude modulation with Type-I embedding to convey more than 1 bit per feature. This is mainly due to the extremely low WNR in these scenarios, which in turn leads to limited dynamic range of detection statistics.

For Type-II, B bits can be embedded in each embedding unit by enforcing a feature derived from this unit into one of $K = 2^B$ subsets. Denoting by I_0 the original image feature, I_1 the watermarked feature, Q the quantization step size, and $m \in \{0, 1, \ldots, K-1\}$ the B-bit information to be embedded, a straightforward extension of odd-even embedding leads to

$$I_1 = \arg \min_{I\ s.t. I = jQ, j \in \mathbf{Z}, \mathrm{mod}(j,K)=m} |I - I_0|. \tag{5}$$

Assuming the distribution of I_0 is approximately uniform over an interval of length KQ, the MSE distortion introduced by embedding is approximately $K^2Q^2/12$. So for a fixed amount of minimal separation Q between the K subsets, a larger K results in larger MSE. And for a fixed amount of allowable embedding distortion, a larger K results in less tolerance to distortion. The idea is easily extensible to table lookup embedding [35] or other enforcement scheme.

Orthogonal and Biorthogonal Modulation: This is mainly used for Type-I embedding. K orthogonal signals are used to represent $B = \log_2 K$ bits by adding one of the K signals to the host media. A detector computes the correlation between the

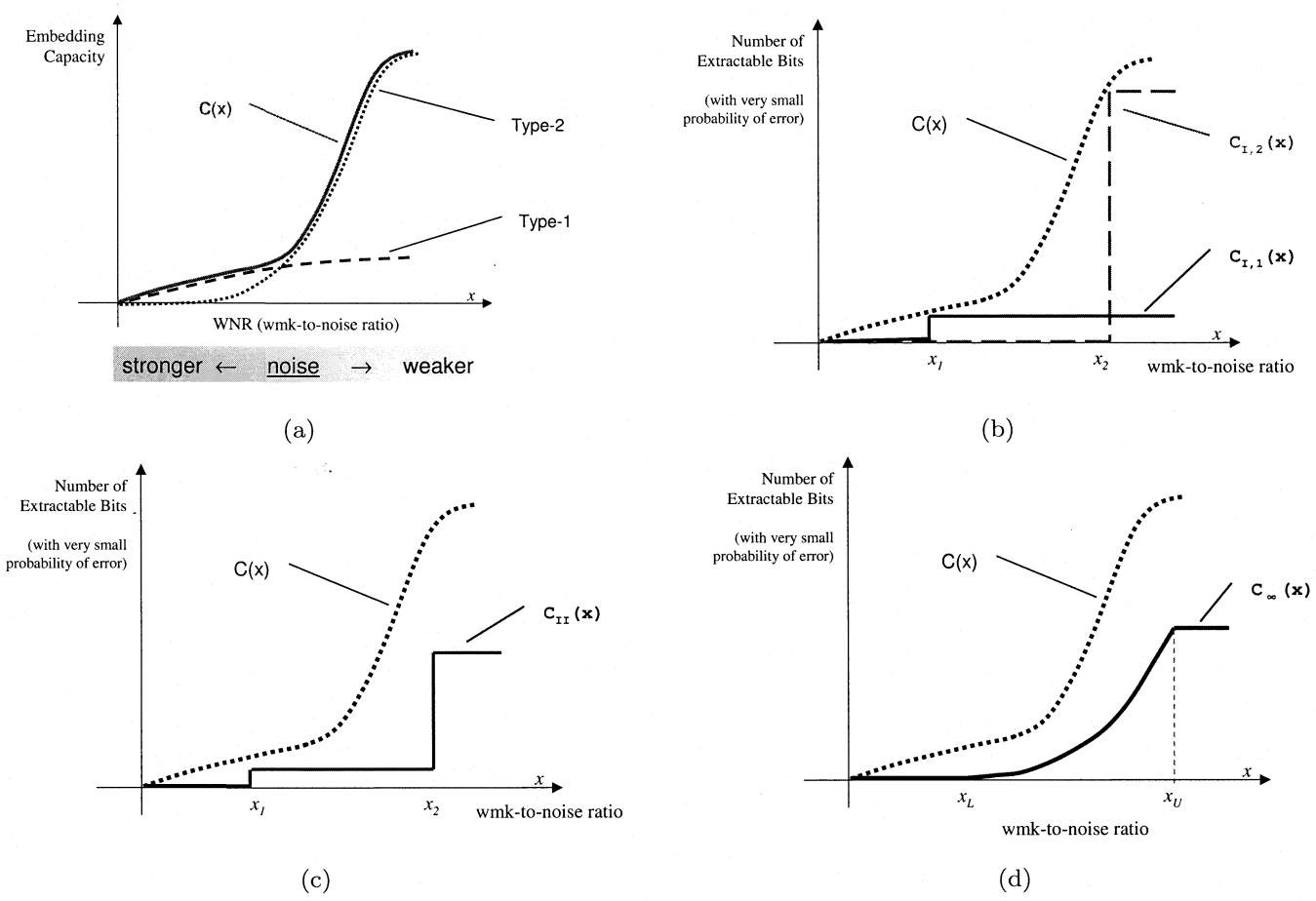

Fig. 5. Amount of extractable data by single-level and multilevel embedding: (a) embedding capacity versus watermark-to-noise ratio, (b)-(d) the number of extractable payload bits by single embedding level, by two embedding levels, and by infinitely many embedding levels, respectively.

TABLE I
COMPARISON OF MODULATION/MULTIPLEXING TECHNIQUES (n ELEMENTS PER EMBEDDING UNIT)

	Amplitude Modulo	*TDM / CDM* ($B \le n$)	Orthogonal ($2^B \le n$)	Biorthogonal ($2^B \le n$)
Type-I		Applicable	Applicable	Applicable
Type-II	Applicable	Applicable		
# embedded bits per element ($\mathcal{X}$)	$\frac{B}{n}$	$\frac{B}{n}$	$\frac{B}{n}$	$\frac{B+1}{n}$
MSE distortion per element ($\mathcal{Y}$)	$\frac{2^{2B}Q^2}{12n}$	$\frac{\mathcal{E}}{n}$	$\frac{\mathcal{E}}{n}$	$\frac{\mathcal{E}}{n}$
minimum separation ($\mathcal{Z}$)	Q	$2\sqrt{\frac{\mathcal{E}}{B}}$	$\sqrt{2\mathcal{E}}$	$\sqrt{2\mathcal{E}}$
$\mathcal{U} = \mathcal{X} \cdot \mathcal{Z}^2/\mathcal{Y}$ energy efficiency	$\frac{12B}{2^{2B}}$	4	$2B$	$2(B+1)$
computational complexity for detecting B bits	constant	$O(B)$	$O(2^B)$	$O(2^B)$

test signal and each of the K signals, and the signal that produces the largest correlation and exceeds a threshold is decided as the embedded signal. A variation, referred to as *biorthogonal modulation*, encodes $\log_2 2K = (B+1)$ bits by adding or subtracting one of K signals [41]. The detection complexity of orthogonal and biorthogonal modulations is exponential with respect to B, and thus inefficient except for small B[8].

[8]A divide-and-conquer detection algorithm for orthogonal modulation recently proposed in [42] can reduce the computation complexity from $O(2^B)$ to $O(B)$ at an expense of detection accuracy.

There is considerable freedom in selecting the K orthogonal signals, but letting the embedder and the decoder agree on the K high-dimensional signals is often nontrivial. In practice, we can use K approximately orthogonal random signals generated from a set of keys, and make the keys known to the embedder and the decoder.

TDM-Type Modulation and Multiplexing: This approach partitions the host media into nonoverlapped segments and hides one or more bits in each segment. TDM-type modulation can be used for both Type-I and Type-II embedding. However, different regions/segments are able to tolerate different amount of changes without causing perceptible artifacts. This uneven embedding capacity can be handled with random shuffling, which will be addressed in Section V.

CDM-Type Modulation and Multiplexing: For Type-I embedding, B bits are encoded into to a watermark signal $\underline{w}$ via

$$\underline{w} = \sum_{k=1}^{B} b_k \cdot \underline{u}_k \tag{6}$$

where $\{\underline{u}_k\}$ are often mutually orthogonal and $b_k \in \{\pm 1\}$. As in orthogonal modulation, there is considerable freedom in selecting $\{\underline{u}_k\}$. But unlike orthogonal modulation, the total signal energy here is the sum of the energy allocated for each bit. For a fixed amount of total energy, the energy per bit is reduced as B increases, causing a decrease in detection reliability.

For Type-II, the embedding of multiple bits can be done by enforcing relations along several mutually orthogonal directions [43], [44].

B. Comparison of Modulation/Multiplexing Techniques

Applicability: The applicability of a particular technique depends on the type of multimedia sources and the embedding mechanism being used. Amplitude modulo modulation is applicable to most medias including audio, image, and video, as long as the features used in embedding are properly chosen. TDM can be used temporally for audio and video, and spatially for image and video. For both general CDM[9] and orthogonal/biorthogonal modulation, one needs to find mutually orthogonal directions in the embedding domain, which can be nontrivial. For example, it is difficult to find in a binary image many overlapped but orthogonal directions to produce features that are manipulable within the just-noticeable-difference (JND) range [53], [54]; to obtain such directions for audio also requires a large window of samples, which could lead to significant processing delay.

TDM versus CDM: TDM and orthogonal CDM are equivalent in terms of energy allocation. TDM is a special case of CDM with the support of $\{\underline{u}_k\}$ nonoverlapping in the sample domain for different k. Alternatively one can choose orthogonal but overlapped $\{\underline{u}_k\}$, as in CDM. The confidentiality of $\{\underline{u}_k\}$ can potentially add an additional layer of security. And uneven embedding capacity is no longer a concern because $\{\underline{u}_k\}$ can be chosen to spread each bit over the entire host signal. However, the B orthogonal sequences have to be generated in CDM and shared with the detector(s), which can be nontrivial for large B. The TDM and CDM approaches can be combined to encode multiple bits.

TDM/CDM versus Orthogonal Modulation: The orthogonal modulation and TDM/CDM-type modulation can be compared by studying the distances between points in signal constellation. This distance is related to detection errors [41]. To convey B bits with a total amount of energy $\mathcal{E}$, the minimum distance between signal points is $\sqrt{2\mathcal{E}}$ for orthogonal modulation, and is $2\sqrt{\mathcal{E}/B}$ for TDM/CDM. So for $B > 2$, orthogonal modulation gives smaller probability of detection error at a cost of complexity in computation and bookkeeping.

By combining orthogonal modulation with TDM or CDM, it can be shown that the embedding rate will increase considerably. In fact, we can double the embedding rate with little complexity increase. For example, the watermark can be constructed as

$$\underline{w} = \sum_{k=1}^{B} b_k \cdot \left[\mathcal{I}(b_{B+k} = 1) \cdot \underline{u}_k^{(1)} + \mathcal{I}(b_{B+k} \neq 1) \cdot \underline{u}_k^{(2)}\right] \tag{7}$$

where $b_i \in \{+1, -1\}$, $\mathcal{I}(\cdot)$ is an indicator function, and all vectors in the two sets $\{\underline{u}_k^{(1)}\}$ and $\{\underline{u}_k^{(2)}\}$ are orthogonal. Here TDM/CDM is used to convey B bits and the orthogonal modulation is used to double the payload. The resulting total watermark energy is the same as using TDM or CDM alone.

Energy Efficiency: The energy efficiency can be compared through $\mathcal{U} = \mathcal{X} \cdot \mathcal{Z}^2/\mathcal{Y}$, where $\mathcal{X}$ is the number of embedded bits per element, $\mathcal{Y}$ the MSE distortion per element introduced by embedding, and $\mathcal{Z}$ the minimum separation between the enforced constellation points. A larger $\mathcal{U}$ value is preferred. As summarized in Table I, except for very small n and B, biorthogonal technique has the largest $\mathcal{U}$ values, while the amplitude modulo technique gives the smallest values (3 for $B = 1$, and 3/2 for $B = 2$). Further, TDM/CDM shows a good balance between energy efficiency and detection complexity as well as broad applicability to both Type-I and Type-II embedding. Examples on using different modulation techniques to hide a nontrivial number of bits in images and video will be presented in Part II.

V. Handling Uneven Embedding Capacity

Changes made in smooth regions of an image are easier to be noticed than those made in textured regions. This leads to unevenly distributed embedding capacity from region to region. We shall refer to a pixel or coefficient of the host media as *embeddable* if it can be modified by more than a predetermined amount without introducing perceptible distortion. The predetermined amount of modification usually depends on both robustness and imperceptibility requirements. For example, a DCT coefficient whose magnitude is smaller than a threshold may be considered as *unembeddable* [12].

To embed as many bits as possible in each region, the number of actually embedded bits would vary significantly from region to region, and this side information has to be conveyed to the detector for decoding. Under blind detection where a detector does not have the original unwatermarked copy, an accurate estimation of how many bits are embedded in each region is not always easy, especially when the watermarked image may have been

[9]TDM can be regarded as a special case of CDM. Here, by "general" we mean to exclude the case of TDM.

subjected to distortion. An error in this estimation can cause not only detection errors in the associated region but also synchronization errors that affect the data extracted from the neighboring regions. Unless the number of bits that can be embedded in each region is large, conveying this side information would introduce large overhead, and may even exceeds the number of bits that can be reliably embedded in the first place.

A common way to overcome this difficulty is to embed a fixed number of bits in each region, thereby eliminating the need of side information. For this approach to work, the fixed number of bits must be small and the size of each region must be large enough to ensure that each region has the capability for embedding this fixed number of bits. Large region size reduces the total number of bits that can be embedded. This approach also causes significant waste in embedding capabilities for regions that are able to hide more bits.

In this section, we propose an adaptive solution to uneven embedding capacity. We use variable rate embedding (VER) if the number of bits that can be embedded is much larger than the number of bits needed to convey how many bits are actually embedded in each region, and we deliver the side information through embedding. If the number of bits that can be embedded is not much larger than the number of bits needed to send the side information, we use constant rate embedding (CER) and show how shuffling can be used to overcome uneven embedding capacity.

A. Quantitative Model for Uneven Embedding Capacity

We consider the blockwise DCT transform of an image of size $S = M_1 \times M_2$, with each transform coefficient labeled as "embeddable" or "unembeddable." The block size of the transform is fixed as 8×8. DC coefficients and the AC coefficients whose magnitude is smaller than a perceptual threshold are left unchanged to avoid artifacts [12], [35]. In a typical natural image such as the one shown in Fig. 6, about 20% of the 8×8 blocks are smooth and have no embeddable coefficients. This is illustrated in Fig. 7.

Suppose n of the S coefficients are embeddable. Then the fraction of embeddable coefficients is $p = n/S$. The coefficients from all blocks can be concatenated into a single string of length S, and this string is divided into N segments of equal length $q = S/N$. Let m_r be the number of segments having r embeddable coefficients, where $r = 0, 1, 2, \ldots, q$. In particular, m_0/N is the fraction of segments having no embeddable coefficients. For the image in Fig. 6 with segment size $q = 8 \times 8 = 64$, the histogram of m_r/N versus r is shown as a solid line in Fig. 8. It is seen that about 20% of the segments have no embeddable coefficients, while a small number of segments have as many as 25 embeddable coefficients. This demonstrates that there can be a large variation in the distribution of embeddables in a natural image. By increasing the segment size q from 64 to 256, a similar shaped histogram is obtained, where the fraction of blocks with no embeddable coefficient is only decreased to 15%. This indicates that to embed a constant number of bits in each segment, simply increasing the segment size is ineffective in reducing the number of segments having zero embeddable coefficients. At the same time, embedding capabilities is wasted in other regions that could potentially hide many bits.

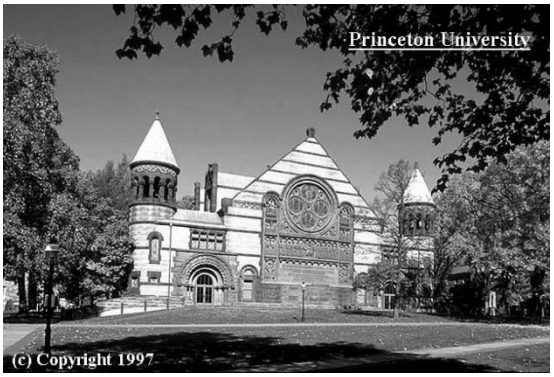

Fig. 6. An original unmarked 640×432 image *Alexander Hall*.

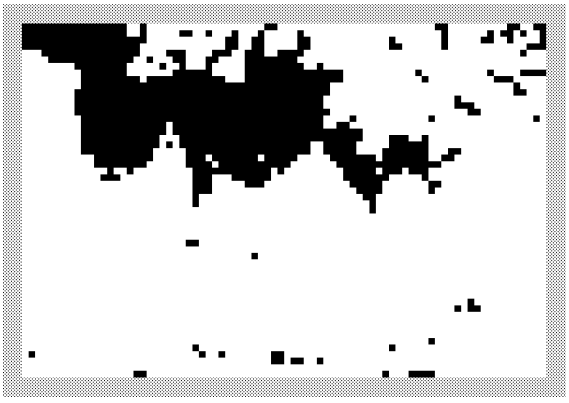

Fig. 7. Smooth blocks of Fig. 6 (shown in black).

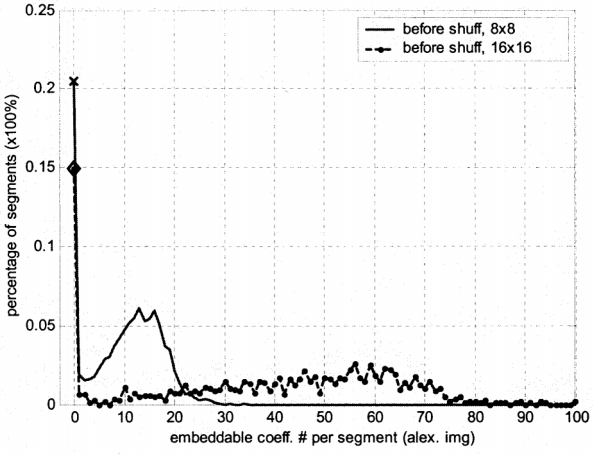

Fig. 8. Histogram of embeddable coefficients per segment for the luminance components of Fig. 6 before shuffling with segment size 8×8 (solid line) and 16×16 (line with dots), respectively.

B. Constant Embedding Rate (CER)

The simplest case of constant embedding rate for either Type-I or Type-II is to embed one bit in each segment. An effective method to overcome uneven distribution of embedding capacity among the blocks is to use shuffling, as illustrated in Fig. 9. The top string is the original one formed by concatenation of all segments. A shuffle is applied to it, resulting in the second string. Embedding is done on this string to produce the third string. For example, the second number

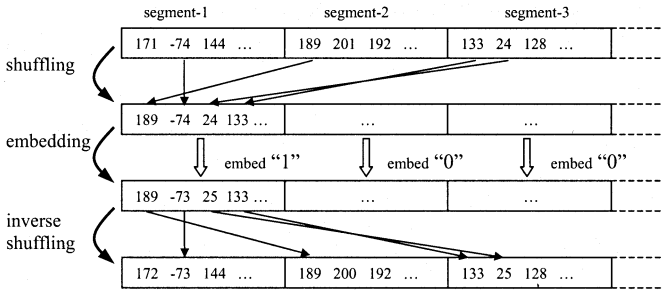

Fig. 9. Incorporate shuffling with an embedding mechanism.

"−74" is changed to "−73," the third number "24" is changed to "25," and so on. The third string is then reversely shuffled to get the fourth string, which is the watermarked signal. The same shuffle needs to be performed at detection.

Shuffling can be considered as a permutation, which can be either random or nonrandom[10]. We shall focus on the case of random permutation, where all permutations are equiprobable [18], [45]. We will show the effectiveness of this approach by examining the distribution of embeddable DCT coefficients before and after a random shuffling.

1) Equalizing Embedding Capacity Via Shuffling: The fraction of segments having r embeddable coefficients is m_r/N. Computing the exact distribution of m_r is quite involved. Instead, we adopt a *moment approach* [46] to study the mean and variance of each normalized bin m_r/N of the histogram. As shall be seen, studying higher moments is not necessary. For each bin m_r of the histogram where $r = 0, \ldots, q$, we have

$$E\left[\frac{m_r}{N}\right] = \frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}} \tag{8}$$

$$Var\left[\frac{m_r}{N}\right] = \frac{1}{N}\cdot\frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}} + \left(1-\frac{1}{N}\right)\frac{\binom{q}{r}\binom{q}{r}\binom{S-2q}{n-2r}}{\binom{S}{n}} - \left[\frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}}\right]^2. \tag{9}$$

The derivation is given in the Appendix .

The expected histogram $\{E(m_i/N)\}$ is a hypergeometric distribution function [47] and can be approximated well by a binomial distribution with mean pq

$$E\left[\frac{m_r}{N}\right] \approx \binom{q}{r} p^r (1-p)^{q-r} \triangleq b(r; q, p) \tag{10}$$

or by Poisson and normal distributions with mean pq. An excellent approximation of $Var[m_r/N]$ is [18]

$$Var\left[\frac{m_r}{N}\right] \approx \frac{1}{N}\cdot b(r; q, p)\cdot[1 - b(r; q, p)]. \tag{11}$$

These quantities depend only on the global parameters p (the percentage of embeddable coefficients) and q (the segment size).

[10]A simple case of nonrandom shuffle is to embed the i-th bit of a total of B bits to $\{kB + i\}$-th coefficients, where k is a positive integer. It is used in such watermarking systems as described in [18] and [35].

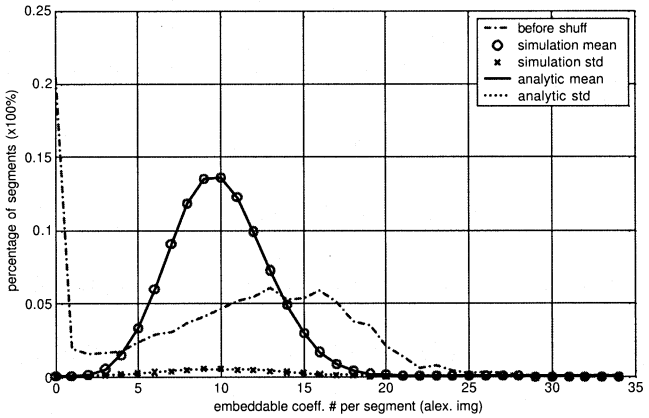

Fig. 10. Histogram of embeddable coefficients per segment for the luminance components of Fig. 6 for block size 8×8 before shuffling (dash-dot line) and after shuffling (all others): (solid line) – mean from analytic study; (dot line) – standard deviation from analytic study; (circles) – mean of simulation; (crosses) – standard deviation of simulation.

We use the image of Fig. 6 as an example, which has a total of $S = 640 \times 432$ block-DCT coefficients, of which $p = 15.49\%$ are embeddable. We choose segment size as $q = 8 \times 8 = 64$, which coincides with the block size of the transform. Equation (8) is plotted as the solid line in Fig. 10. The square root of (9), or the standard deviation, is the dotted line in the same figure. For comparison, the dash-dot line in Fig. 10 is the histogram of $\{m_r/N\}$ before shuffling, showing that 20% of the segments have no embeddable coefficients. The very small value of $Var[m_r/N]$ after shuffle suggests that very few shuffles will result in a histogram which deviates appreciably from the mean, and that higher moments will not contribute much to this investigation.

The values from (8) and (9) are

$$E\left[\frac{m_0}{N}\right] \approx 0.002\%, \quad Var\left[\frac{m_0}{N}\right] \approx 4.85 \times 10^{-9}. \tag{12}$$

The mean value indicates that the average fraction of segments with no embeddable coefficients is reduced by 4 orders of magnitude, from the original 20% to 0.002%. The expected number of blocks with no embeddable coefficient after shuffling is only

$$0.002\% \times N = 0.002\% \times 640 \times 432/64 \approx 0.086. \tag{13}$$

We have also performed 1000 random permutations on the block-DCT coefficients of the image of Fig. 6. The sample mean and sample variance of each bin of the histogram m_r/N are also shown in Fig. 10. The circles are the average fraction of segments having a given number of embeddable coefficients from simulation, and the crosses are the standard deviation from simulation. The agreement of simulation and analysis is excellent. We also see that after shuffling, the number of segments which have no embeddable coefficients has been significantly reduced and that most segments have between 5 and 15 embeddable coefficients.

It should be pointed out that q does not have to be the same as the block size of the transform (8×8). Instead, q should

be chosen to produce the desired mean, pq, of the histogram $\{E(m_i/N)\}$, and to make sure that the left tail of the histogram is smaller than a desired bound. For images that contain a large fraction of embeddable coefficients (i.e., large p), the segment size can be small; while for images in which most regions are smooth, the segment size should be large enough to ensure sufficient decay at the left tail.

Shuffling in most cases will produce at least one embeddable coefficient in all segments, allowing one bit to be embedded in every shuffled segment. It is noted that the embedding in smooth regions via shuffling is in a *logical sense*. No bits are actually inserted in smooth regions. Instead, many embeddable coefficients from nonsmooth regions are dynamically allocated through shuffling to hold the data that are intended to be put in smooth regions. Also, shuffling will not compromise perceptual quality, as long as the criterion for identifying embeddable coefficients is unchanged.

The equalization of embedding capacity via shuffling does require a little additional side information. The detector only needs to know the segment size and the shuffle table. A shuffling table can be generated efficiently from a key with linear complexity proportional to the number of entries [48].

2) Practical Considerations: While our analysis shows that the probability for getting a bad shuffle that cannot equalize the uneven capacity is extremely small, it is still possible for a given image. When all but very few blocks having no embeddable coefficients, the bits to be embedded in the blocks with no embeddables can be treated by a detector as *erasure bits*. Applying error correction coding [49] with moderate correction capability to the data to be embedded will be able to handle this problem. Another approach to handle bad shuffle is to generate a set of candidate shuffles which are significantly different from each other, and use the best shuffle for a given image. Since the probability that a number of shuffles are all bad decreases exponentially from the already low probability of a single shuffle, two candidate shuffles would be adequate in practice. We can use one as a primary shuffle, and switch to a secondary one when the primary one is not suitable for a given image. How to convey to the detector on which shuffle is used is similar to conveying side information in variable rate embedding, and will be discussed further in Section V-C. Additionally, the segment size q determines how many bits will be embedded in an image and is dependent on the percentage of embeddable pixels p. If p is small, the segment size has to be large to ensure that a sufficient number of embeddable coefficients are present in each shuffled segment. Because p can vary considerably from image to image, it is desirable to develop a variable segment size strategy. This will be discussed further in Section V-C.

Shuffling increases the sensitivity to such intentional attacks as geometric distortion. This sensitivity can be alleviated through registration with respect to a known reference watermark [50], [51]. Besides applying shuffling to the entire set of samples or coefficients, we can shuffle on block basis by permuting all samples/ coefficients in the same block as a whole [52]. We can also apply different shuffles to each frequency band of a block-based transform so that the coefficients of a particular frequency band remains in their original frequency band but permuted to different blocks.

C. Variable Embedding Rate (VER)

Compared with CER, VER allows more data to be embedded if the average overhead for side information is relatively small compared with the average embedding capacity per segment. An important issue to be addressed is how to convey the side information of the number of bits embedded in each segment. Here we consider a more general problem of sending side information, which facilitates the extractions of the embedded data payload. The side information could be the number of bits being embedded in each segment, or an index indicating which shuffle and/or what segment size is used in the constant-rate embedding (Section V-B2).

The side information can be conveyed either using the same embedding mechanism as for the user payload or using different embedding mechanisms. In both cases, the side information consumes part of the energy by which the host image can be changed imperceptibly. The difference lies only on the specific way to achieve orthogonality, similar to the discussion of TDM/CDM multiplexing and orthogonal modulation in Section IV. Allocating more energy to the side information gives higher robustness in extracting them but reduces the amount of user payload. It is desirable to both limit the amount of side information and use energy efficient modulation techniques to embed multiple bits of side information.

Consider first the embedding of side information via the same embedding mechanism as that for the user payload. We can use a strategy similar to the training sequence in digital communications. That is, part of the embedded data are pre-determined, or designed to be self-verifiable, which can be obtained by hash function (message digest function) or error detection/correction coding. For example, in order to let a detector know which shuffle is used for each image, one may choose some beginning bits of embedded data to be a predetermined label, or a label plus its hash. The detector then decodes the hidden data using all candidate shuffles, and the shuffle that leads to accurately decoding is identified as the one used by the embedder. When we decode the embedded data using a shuffle table that is significantly different from the one used by the embedder, the decoded bits are approximately independent of each other and equiprobable to be "1" or "0." Hence the probability of wrongly identifying which shuffle is being used decreases exponentially with the number of verification bits. Similarly, to let decoder know what segment size is used by the embedding process, we can select from a finite number of candidate segment sizes, and make part of the embedded data pre-determined or self-verifiable. A detector will try out candidate segment sizes and find the one that successfully passes the verification. To limit the search complexity, the segment size suitable for a large number of images can be used as primary, and a few other sizes can be chosen as secondary to handle special images. These strategies of conveying side information have been applied to data hiding in binary images [53], [54].

For grayscale/color images and videos, it is possible to find some other domains or mechanisms for sending side information. The popular spread spectrum additive embedding is one candidate because their statistical properties make it easy to gen-

erate additional "watermarks" orthogonal or approximately orthogonal to the watermarks used for conveying user payload. Spread spectrum embedding can also help to convey side information robustly.

VI. Conclusions

A number of fundamental problems of data hiding have been investigated and solutions proposed. The analysis of the tradeoff of embedding capacity versus robustness for two major types of embedding mechanisms leads to a new, multilevel data embedding framework, which allows the amount of extractable data being adaptive to the actual noise condition. We have also studied the suitable conditions of using various modulation/multiplexing techniques for hiding multiple bits. Finally, we proposed solutions to the problem of unevenly distributed embedding capacity. The choice between a constant embedding rate and a variable embedding rate depends on the overhead of side information relative to the total embedding payload. For constant embedding rate, shuffling is proposed to dynamically equalize the distribution of embeddable coefficients, allowing for hiding more data. For variable embedding rate, we have shown how to convey side information to ensure the correct detection of the embedded user payload. These results are applied to data hiding algorithms for image and video in Part II [1].

Appendix

We present the derivation of (8) and (9) on shuffling in this appendix. Considering a bin m_r of the embeddability histogram where r is an integer between 0 and q, we perform the following decomposition

$$m_r = \theta_{r,1} + \theta_{r,2} + \cdots + \theta_{r,N} \tag{14}$$

where $\theta_{r,i}$ is an indicator function defined as

$$\theta_{r,i} = \begin{cases} 1, & i^{th} \text{ segment has } r \text{ embeddable coeff.} \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Computing the mean of $\theta_{r,i}$ is equivalent to getting the probability that the i^{th} segment has r embeddables, i.e.,

$$E[\theta_{r,i}] = P[\theta_{r,i} = 1] = \frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}}. \tag{16}$$

Since the mean of θ_i is independent of i, we have

$$E\left[\frac{m_r}{N}\right] = E\left[\frac{\sum_{i=1}^{N}\theta_{r,i}}{N}\right] = E(\theta_{r,1}) = \frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}}. \tag{17}$$

This is the result of (8). The variance is derived by observing ${\theta_{r,i}}^2 = \theta_{r,i}$, from which we have

$${m_r}^2 = \sum_{k=1}^{N}{\theta_{r,k}}^2 + \sum_{i\neq j}\theta_{r,i}\theta_{r,j} = m_r + \sum_{i\neq j}\theta_{r,i}\theta_{r,j}. \tag{18}$$

For $i \neq j$

$$E[\theta_{r,i}\theta_{r,j}] = P(\theta_{r,i} = \theta_{r,j} = 1) = \frac{\binom{q}{r}\binom{q}{r}\binom{S-2q}{n-2r}}{\binom{S}{n}} \tag{19}$$

which is independent of i and j. We now have

$$E[{m_r}^2] = E[m_r] + E\left[\sum_{i\neq j}\theta_{r,i}\theta_{r,j}\right] \tag{20}$$

$$= E[m_r] + N(N-1)E[\theta_{r,1}\theta_{r,2}]. \tag{21}$$

Therefore, we obtained the result of (9)

$$Var\left[\frac{m_r}{N}\right] = \frac{1}{N^2}E[{m_r}^2] - \left[E\left(\frac{m_r}{N}\right)\right]^2 \tag{22}$$

$$= \frac{1}{N}\cdot\frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}} + \left(1-\frac{1}{N}\right)\frac{\binom{q}{r}\binom{q}{r}\binom{S-2q}{n-2r}}{\binom{S}{n}} - \left[\frac{\binom{q}{r}\binom{S-q}{n-r}}{\binom{S}{n}}\right]^2. \tag{23}$$

Acknowledgment

The authors would like to thank Prof. P. Diaconis of Stanford University and Profs. B. Dickinson, E. Cinlar, and S. Kulkarni of Princeton University for recommending helpful references on random allocation.

References

[1] M. Wu, H. Yu, and B. Liu, "Data hiding in image and video: Part II—Designs and applications," *IEEE Trans. Image Processing*, this issue.
[2] I. J. Cox, M. L. Miller, and J. A. Bloom, *Digital Watermarking*. San Mateo, CA: Morgan Kaufmann, 2001.
[3] I. J. Cox and M. L. Miller, "The first 50 years of electronic watermarking," *EURASIP J. Appl. Signal Process.*, vol. 2002, no. 2, pp. 126–132, Feb. 2002.
[4] F. Mintzer, G. W. Braudaway, and M. M. Yeung, "Effective and ineffective digital watermarks," in *IEEE Int. Conf. Image Processing (ICIP'97)*, 1997.
[5] R. J. Anderson and F. A. P. Petitcolas. (1999) Information Hiding: An Annotated Bibliography. [Online]. Available: http://www.cl.cam.ac.uk/~fapp2/ steganography/ bibliography/
[6] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding—A survey," *Proc. IEEE*, pp. 1062–1078, July 1999.
[7] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proc. IEEE*, pp. 1079–1107, July 1999.
[8] "Special issue on watermarking," *Signal Process.*, vol. 66, no. 3, May 1998.
[9] "Special issue on watermarking," *Commun. ACM*, vol. 41, no. 7, July 1998.
[10] "Copyright and privacy protection, special issue," *IEEE J. Select. Areas Commun.*, vol. 16, May 1998.
[11] "Special issue on identification and protection of multimedia information," *Proc. IEEE*, July 1999.
[12] W. Zeng, "Resilient Video Transmission and Multimedia Database Application," Ph.D. dissertation, Princeton Univ., Princeton, NJ, 1997.
[13] L. Qiao, "Multimedia security and copyright protection," Ph.D. dissertation, Univ. Illinois at Urbana-Champaign, 1998.
[14] D. Kundur, "Multiresolution digital watermarking: Algorithms and implications for multimedia signals," Ph.D. dissertation, Univ. Toronto, Toronto, ON, Canada, 1999.
[15] M. Ramkumar, "Data hiding in multimedia—Theory and applications," Ph.D. dissertation, New Jersey Inst. Technol., Newark, 2000.
[16] B. Chen, "Design and analysis of digital watermarking, information embedding, and data hiding systems," Ph.D. dissertation, MIT, Cambridge, MA, 2000.
[17] C.-Y. Lin, "Watermarking and digital signature techniques for multimedia authentication and copyright protection," Ph.D. dissertation, Columbia Univ., New York, 2000.
[18] M. Wu, "Multimedia data hiding," Ph.D. dissertation, Princeton Univ., Princeton, NJ, 2001.
[19] W. Bender, D. Gruhl, and N. Morimoto, "Techniques for data hiding," *Proc. SPIE*, vol. 2420, 1995.

[20] I. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing*, vol. 6, pp. 1673–1687, Dec. 1997.
[21] J. R. Smith and B. O. Comiskey, "Modulation and information hiding in images," in *Proc. 1st Information Hiding Workshop*, 1996.
[22] S. D. Servetto, C. I. Podilchuk, and K. Ramchandran, "Capacity issues in digital image watermarking," in *IEEE Int. Conf. Image Processing (ICIP'98)*, Chicago, IL, 1998.
[23] M. Ramkumar and A. N. Akansu, "Information theoretic bounds for data hiding in compressed images," in *Proc. IEEE 2nd Multimedia Signal Processing Workshop*, 1998.
[24] L. M. Marvel and C. G. Boncelet. (1999) Capacity of the additive steganographic channel. [Online]. Available: http://www.eecis.udel.edu/~marvel/
[25] M. Barni, F. Bartolini, A. De Rosa, and A. Piva, "Capacity of full frame DCT image watermarks," *IEEE Trans. Image Processing*, vol. 9, pp. 1450–1455, Aug. 2000.
[26] C. E. Shannon, "The zero-error capacity of a noisy channel," *IRE Trans. Inform. Theory*, vol. IT-2, pp. 8–19, 1956.
[27] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York: Wiley, 1991.
[28] M. H. M. Costa, "Writing on dirty paper," *IEEE Trans. Inform. Theory*, vol. IT-29, May 1983.
[29] P. Moulin and J. A. O'Sullivan. (1999) Information-theoretic analysis of information hiding. [Online]. Available: http://www.ifp.uiuc.edu/~moulin/paper.html
[30] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inform. Theory*, vol. 47, pp. 1423–1443, May 2001.
[31] M. Wu, H. Yu, and A. Gelman, "Multi-level data hiding for digital image and video," in *Proc. Photonics East Conference on Multimedia Systems and Applications*, vol. 3845, Boston, MA, Sept. 1999.
[32] W. Zeng and B. Liu, "A statistical watermark detection technique without using original images for resolving rightful ownerships of digital images," *IEEE Trans. Image Processing*, vol. 8, pp. 1534–1548, Nov. 1999.
[33] C. Podilchuk and W. Zeng, "Image adaptive watermarking using visual models," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 525–539, May 1998.
[34] M. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *IEEE Int. Conf. Image Processing (ICIP'97)*, 1997.
[35] M. Wu and B. Liu, "Watermarking for image authentication," in *IEEE Int. Conf. Image Processing (ICIP'98)*, Chicago, IL, 1998.
[36] E. Koch and J. Zhao, "Toward robust and hidden image copyright labeling," in *IEEE Workshop on Nonlinear Signal and Image Processing*, 1995.
[37] C.-T. Hsu and J.-L. Wu, "Hidden signatures in image," in *IEEE Int. Conf. Image Processing (ICIP'96)*, vol. 3, 1996.
[38] M. Ramkumar and A. N. Akansu, "A robust scheme for oblivious detection of watermarks/data hiding in still images," in *Proc. Symp. Voice, Video, and Data Communication*, 1998.
[39] M. Kesal, M. K. Mihcak, R. Koetter, and P. Moulin, "Iteratively decodable codes for watermarking applications," in *Proc. 2nd Int. Symp. Turbo codes and Related Topics*, Brest, France, Sept. 2000.
[40] J. J. Eggers, R. Bauml, R. Tzschoppe, and B. Girod. (2002) Scalar Costa scheme for information embedding. [Online]. Available: http:// www-nt.e-technik.uni-erlangen.de/~eggers/ publications.html
[41] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995.
[42] W. Trappe, M. Wu, Z. Wang, and K. J. R. Liu, "Anti-collusion fingerprinting for multimedia," *IEEE Trans. Signal Processing*, vol. 51, pp. 1069–1087, Apr. 2003.
[43] M. D. Swanson, B. Zhu, and A. H. Tewfik, "Robust data hiding for images," in *Proc. IEEE DSP Workshop*, 1996.
[44] M. Alghoniemy and A. H. Tewfik, "Self-synchronizing watermarking techniques," in *Proc. Symp. Content Security and Data Hiding in Digital Media*, 1999.
[45] M. Wu and B. Liu, "Digital watermarking using shuffling," in *Proc. IEEE Int. Conf. Image Processing (ICIP'99)*, Kobe, Japan, 1999.
[46] V. F. Kolchin, B. A. Sevastyanov, and V. P. Chistyakov, *Random Allocation*. London, U.K.: Winston, 1978.
[47] W. Feller, *An Introduction to Probability Theory and Its Applications*, 3rd ed. New York: Wiley, 1970, vol. 1.
[48] D. E. Knuth, *The Art of Computer Programming*, 3rd ed. Reading, MA: Addison-Wesley, 1997, vol. 2.
[49] R. E. Blahut, *Theory and Practice of Data Transmission Codes*, 2nd ed., 1997.
[50] S. Pereira and T. Pun, "Fast robust template matching for affine resistant image watermarks," in *Proc. 3rd Information Hiding Workshop (IHW)*, Lecture Notes in Computer Science, 1999, pp. 207–218.
[51] G. Csurka, F. Deguillaume, J. J. K. ÓRuanaidh, and T. Pun, "A Bayesian approach to affine transformation resistant image and video watermarking," in *Proc. 3rd Information Hiding Workshop (IHW)*, Lecture Notes in Computer Science, 1999, pp. 315–330.
[52] G. C. Langelaar and R. L. Lagendijk, "Optimal differential engergy watermarking of DCT encoded images and video," *IEEE Trans. Image Processing*, vol. 10, pp. 148–158, Jan. 2001.
[53] M. Wu, E. Tang, and B. Liu, "Data hiding in digital binary image," in *IEEE International Conference on Multimedia & Expo (ICME'00)*, New York, 2000.
[54] M. Wu and B. Liu, "Data hiding in binary images," *IEEE Trans. Multimedia*, 2003, to be published.

Min Wu (S'95–M'01) received the B.E. degree in electrical engineering and the B.A. degree in economics from Tsinghua University, Beijing, China, in 1996 (both with the highest honors), and the M.A. degree and Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, in 1998 and 2001, respectively.

She was with NEC Research Institute and Signafy, Inc., in 1998, and with Panasonic Information and Networking Laboratories in 1999. Since 2001, she has been an Assistant Professor of the Department of Electrical and Computer Engineering, the Institute of Advanced Computer Studies, and the Institute of Systems Research at the University of Maryland, College Park. Her research interests include information security, multimedia signal processing, and multimedia communications.

Dr. Wu received a CAREER award from the U.S. National Science Foundation in 2002 and holds three U.S. patents on multimedia data hiding. She is a member of the IEEE Technical Committee on Multimedia Signal Processing and Publicity Chair of 2003 IEEE International Conference on Multimedia and Expo (ICME'03, Baltimore, MD). She co-authored a book *Multimedia Data Hiding* (New York: Springer-Verlag, 2002), and is a Guest Editor of special issue on Multimedia Security and Rights Management of the *Journal on Applied Signal Processing*.

Bede Liu (S'55–M'62–F'72) was born in China and studied at the National Taiwan University (B.S.E.E., 1954) and the Polytechnic Institute of Brooklyn (D.E.E., 1960).

Prior to joining Princeton University, Princeton, NJ, in 1962, he had been with Bell Laboratories, Allen B. DuMont Laboratory, and Western Electric Company. He has also been a visiting faculty member at several universities in U.S. and abroad. He is Professor of electrical engineering at Princeton University. His current research interest lies mostly with multimedia technology, with particular emphasis on digital watermarking and video processing.

Dr. Liu and his students have twice received the Best Paper Awards on Video Technology (1994 and 1996). His other IEEE awards include Centennial Medal (1984) and Millennium Medal (2000), Signal Processing Society's Technical Achievement Award (1985) and Society Award (2000), Circuit and Systems Society's Education Award (1988) and Mac Van Valkenburg Award (1997). He is a member of the National Academy of Engineering. He was the President of the Circuit and Systems Society (1982), and the IEEE Division I Director (1984, 1985). He also served as the 1978 ISCAS Technical Program Chair and the General Chair of the 1995 ICIP.